AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Previously presented) A method for checking a model, which defines states of a system under study and a transition relation among the states, the method comprising:

specifying a property that applies to a target set that comprises at least one target state among the states of the system under study;

beginning from an initial set of at least one initial state among the states of the system, computing successive reachable sets comprising the states of the system that are reachable from the initial set;

finding an intersection between one of the reachable sets and the target set;

computing a plurality of mutually-disjoint traces from the at least one target state in the intersection through the states in the reachable sets to the at least one initial state, wherein the plurality of mutually-disjoint traces comprises at least one initial trace and at least one subsequently-computed trace;

selecting the states on each subsequently-computed trace so as to maximize a distance of the subsequently-computed trace from at least the at least one initial trace; and

using the computed traces for checking the model.

2. (Original) A method according to claim 1, wherein specifying the property comprises specifying a condition that is expected to be true over all of the reachable states of the system under study, and wherein the condition is false in the at least one target state.

3. (Original) A method according to claim 1, wherein specifying the property comprises specifying a condition representing a desired behavior of the system under study, such that the condition is fulfilled in the at least one target state.

4. (Previously presented) A method according to claim 1, wherein computing the successive reachable sets comprises testing the property while computing the successive reachable sets, and ceasing to compute the successive reachable sets when the intersection is found.

5. (Original) A method according to claim 1, wherein computing the successive reachable sets comprises:

determining a first one of the reachable sets, disjoint from the initial set, such that all of the states in the first

one of the reachable sets are reached from the at least one initial state in a first cycle of the transition relation; and

determining the successive reachable sets, following the first one of the reachable sets, such that all of the states in each of the reachable sets are reached from the states in a preceding one of the reachable sets in a successive cycle of the transition relation, and so that each of the successive reachable sets is disjoint from the initial set and from the other reachable sets determined before it.

6. (Previously presented) A method according to claim 5, wherein computing the traces comprises, for each trace among the plurality of mutually-disjoint traces, selecting one of the states from each of the successive reachable sets.

7. (Currently amended) A method according to claim 6, wherein selecting the one of the states comprises, for each of the ~~selected~~ states that is selected, choosing a predecessor state among the states in a preceding one of the reachable sets until the state on the trace in the first one of the reachable sets is found, and choosing the predecessor state in the initial set to be the state in the first one of the reachable sets.

8. (Previously presented) A method according to claim 7, wherein choosing the predecessor state comprises, on each

of the traces computed after a first one of the traces, choosing the predecessor state so as to maximize the distance of the trace from the other traces already computed.

9. (Canceled)

10. (Currently amended) A method according to claim 1, wherein each of the states is represented by a binary decision diagram (BDD), and wherein selecting the states on each trace comprises maximizing the distance between the BDD representing the state to be selected and the BDD representing the states on the other traces already computed.

11. (Original) A method according to claim 10, wherein maximizing the distance comprises:

taking a left trial state and a right trial state on left and right branches, respectively, of the BDD representing the state to be selected; and

choosing the trial state that has a larger Hamming distance from the BDD representing the states on the other traces.

12. (Previously presented) Model checking apparatus, comprising a model processor, which is arranged to receive a model defining states of a system under study and a transition relation among the states, and to receive a specification of a property that applies to a target set comprising at least one

target state among the states of the system under study, the processor being further arranged to compute, beginning from an initial set of at least one initial state among the states of the system, successive reachable sets comprising the states of the system that are reachable from the initial set, to find an intersection between one of the reachable sets and the target set, and to compute, for use in checking the model, a plurality of mutually-disjoint traces from the at least one target state in the intersection through the states in the reachable sets to the at least one initial state, wherein the plurality of mutually-disjoint traces comprises at least one initial trace and at least one subsequently-computed trace, and wherein the states on each subsequently-computed trace are selected so as to maximize a distance of the subsequently-computed trace from at least the at least one initial trace.

13. (Original) Apparatus according to claim 12, wherein the property comprises a condition that is expected to be true over all of the reachable states of the system under study, and wherein the condition is false in the at least one target state.

14. (Original) Apparatus according to claim 12, wherein the property comprises a condition representing a desired behavior of the system under study, such that the condition is fulfilled in the at least one target state.

15. (Previously presented) Apparatus according to claim 12, wherein the processor is arranged to test the property while computing the successive reachable sets, and ceases to compute the successive reachable sets when the intersection is found.

16. (Original) Apparatus according to claim 12, wherein the processor is arranged to compute a first one of the reachable sets, disjoint from the initial set, such that all of the states in the first one of the reachable sets are reached from the at least one initial state in a first cycle of the transition relation, and further to compute the successive reachable sets, following the first one of the reachable sets, such that all of the states in each of the reachable sets are reached from the states in a preceding one of the reachable sets in a successive cycle of the transition relation, and so that each of the successive reachable sets is disjoint from the initial set and from the other reachable sets determined before it.

17. (Previously presented) Apparatus according to claim 16, wherein the processor is arranged to compute the traces by selecting, for each trace among the plurality of mutually-disjoint traces, one of the states from each of the successive reachable sets.

18. (Previously presented) Apparatus according to claim 17, wherein the processor is arranged to compute the traces by choosing, for each of the selected states, a predecessor state among the states in a preceding one of the reachable sets until the state on the trace in the first one of the reachable sets is found, and choosing the predecessor state in the initial set to be the state in the first one of the reachable sets.

19. (Previously presented) Apparatus according to claim 18, wherein the processor is arranged to choose the predecessor state on each of the traces computed after a first one of the traces so as to maximize the distance of the trace from the other traces already computed.

20. (Canceled)

21. (Currently amended) Apparatus according to claim 12, wherein each of the states is represented by a binary decision diagram (BDD), and wherein the processor is arranged to select the states on each trace so as to maximize the distance between the BDD representing the state to be selected and the BDD representing the states on the other traces already computed.

22. (Original) Apparatus according to claim 21, wherein the processor is arranged to maximize the distance by taking a

left trial state and a right trial state on left and right branches, respectively, of the BDD representing the state to be selected, and choosing the trial state that has a larger Hamming distance from the BDD representing the states on the other traces.

23. (Previously presented) A computer software product, comprising a computer-readable medium in which program instructions are stored, which instructions, when read by a computer, cause the computer to receive a model defining states of a system under study and a transition relation among the states, and to receive a specification of a property that applies to a target set comprising at least one target state among the states of the system under study, the instructions further causing the computer to compute, beginning from an initial set of at least one initial state among the states of the system, successive reachable sets comprising the states of the system that are reachable from the initial set, to find an intersection between one of the reachable sets and the target set, and to compute, for use in checking the model, a plurality of mutually-disjoint traces from the at least one target state in the intersection through the states in the reachable sets to the at least one initial state, wherein the plurality of mutually-disjoint traces comprises at least one initial trace and at least one subsequently-computed trace,

and wherein the states on each subsequently-computed trace are selected so as to maximize a distance of the subsequently-computed trace from at least the at least one initial trace.

24. (Original) A product according to claim 23, wherein the property comprises a condition that is expected to be true over all of the reachable states of the system under study, and wherein the condition is false in the at least one target state.

25. (Original) A product according to claim 23, wherein the property comprises a condition representing a desired behavior of the system under study, such that the condition is fulfilled in the at least one target state.

26. (Previously presented) A product according to claim 23, wherein the instructions cause the computer to test the property while computing the successive reachable sets, and to cease to compute the successive reachable sets when the intersection is found.

27. (Original) A product according to claim 23, wherein the instructions cause the computer to compute a first one of the reachable sets, disjoint from the initial set, such that all of the states in the first one of the reachable sets are reached from the at least one initial state in a first cycle of the transition relation, and further to compute the

successive reachable sets, following the first one of the reachable sets, such that all of the states in each of the reachable sets are reached from the states in a preceding one of the reachable sets in a successive cycle of the transition relation, and so that each of the successive reachable sets is disjoint from the initial set and from the other reachable sets determined before it.

28. (Previously presented) A product according to claim 27, wherein the instructions cause the computer to compute the traces by selecting, for each trace among the plurality of mutually-disjoint traces, one of the states from each of the successive reachable sets.

29. (Previously presented) A product according to claim 28, wherein the instructions cause the computer to compute the traces by choosing, for each of the selected states, a predecessor state among the states in a preceding one of the reachable sets until the state on the trace in the first one of the reachable sets is found, and choosing the predecessor state in the initial set to be the state in the first one of the reachable sets.

30. (Previously presented) A product according to claim 29, wherein the instructions cause the computer to choose the predecessor state on each of the traces computed after a first

one of the traces so as to maximize the distance of the trace from the other traces already computed.

31. (Canceled)

32. (Currently amended) A product according to claim 23, wherein each of the states is represented by a binary decision diagram (BDD), and wherein the instructions cause the computer to select the states on each trace so as to maximize the distance between the BDD representing the state to be selected and the BDD representing the states on the other traces already computed.

33. (Previously presented) A product according to claim 32, wherein the instructions cause the computer to maximize the distance by taking a left trial state and a right trial state on left and right branches, respectively, of the BDD representing the state to be selected, and choosing the trial state that has a larger Hamming distance from the BDD representing the states on the other traces.